

Customer Notification

V850E/Dx3 - DJ3/DL3

32-Bit Single-Chip Microcontrollers

Operating Precautions

 $\begin{array}{c} \mu PD70F3421 \\ \mu PD70F3422 \\ \mu PD70F3423 \\ \mu PD70F3424 \\ \mu PD70F3425 \\ \mu PD70F3426 \\ \mu PD70F3426A \\ \mu PD70F3427 \end{array}$

Notice

- 1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
- 2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
- 3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
- 4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
- 5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
- 6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
- 7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
- 8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
- 9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
- 10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
- 11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
- 12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.
- (Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majorityowned subsidiaries.
- (Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

Table of Contents

| (A) | (A.1) (A.2) | μPD70 μPD70 | of Operating Precautions |
|-----|----------------|----------------|---|
| | (A.3) | μPD70 | F3426, µPD70F3426A, µPD70F3427 |
| (B) | | Descri | ption of Operating Precautions |
| | | No. 1 | Flash Memory: Flash memory block operations (Technical Limitation) |
| | | No. 2 | CPU: RAM mirror access (μPD70F3425 only) (Specification change notice) |
| | | No. 3 | Ports: PDSCn registers mirrors (Specification change notice) |
| | | No. 4 | Code Protection and Security: Limitation of flash memory extra area and reset vector reprogramming (Specification change notice) |
| | | No. 5 | External Interrupt - Input Filter Timing Port P07/ INTP6: Shortened pulse rejection time |
| | | No. 6 | External Interrupt - Input Filter Timing - All external interrupt input channels (except Port P07/ INTP6): Shortened pulse rejection time |
| | | No. 7 | AFCAN: Sleep Mode Wakeup (Specification change notice) |
| | | No. 8 | On-Chip Debug Unit: Maximum input clock for the Debug Control Unit (DCU) is limited to 10 MHz |
| | | No. 9 | On-Chip Debug Unit: Execution of BROM-Startup Code can be unintentionally interrupted at User S/W Breakpoint Addresses |
| | | | CSIB: CSIB may stop operating |
| | | No. 11 | On-Chip Debug Unit: Reconfiguration of N-Wire pins during N-Wire debug mode (µPD70F3427 only) |
| | | No. 12 | Flash Memory: Increased Power Consumption when CPU is operating on the Sub Oscillator Clock (µPD70F3426 only) |
| | | | Flash Memory: Increased Power Consumption when CPU is operating on the Sub Oscillator Clock (iRAM operation) |
| | | | Flash Memory: Flash Access Timing Violation after System Clock Supply Switch 22 |
| | | No. 15 | CPU: HALT instruction (1) (µPD70F3426 only) (Specification change notice) |
| | | No. 16 | CPU: HALT instruction (2) (Specification change notice) |
| | | | DMA: MLE Bit Usage (Specification change notice) |
| | | No. 18 | Linear code execution across address 0x00010000 and 0x00008000 when the boot sectors are swapped (Direction of use) |
| | | No. 19 | CPU: Speed limitation (1) (µPD70F3426 only) (Technical limitation) |
| | | No. 20 | CPU: Speed limitation (2) (µPD70F3427 only) (Technical limitation) |
| | | No. 21 | LCD-I/F: Reading the LBCTL0.TPF0 status flag may result in wrong information (Specification change notice) |
| | | No. 22 | LCD-I/F: Writing to the LBDATA0W/ LBDATA0/ LBDATA0L register may lead to a corrupt data transfer |
| | | | (Technical limitation) |
| (C) | | Valid 9 | Specification |

(D) Revision History......35

(A) Table of Operating Precautions

(A.1) $\mu PD70F3421$, $\mu PD70F3422$, $\mu PD70F3423$

| No. | Outline | | µPD70F3421 | µPD70F3422 | µPD70F3423 | |
|-----|--|----------------------|------------|------------|------------|----------|
| | | Rev. | 1.1 | 1.1 | 1.0 | 1.1 |
| | | Rank ^{Note} | I | Z | Ι | K |
| 1 | Flash Memory: Flash memory block operations Limitation) | s (Technical | 1 | 1 | × | ✓ |
| 2 | CPU: RAM mirror access (μPD70F3425 only) (Schange notice) | Specification | ✓ | 1 | 1 | ✓ |
| 3 | Ports: PDSCn registers mirrors (Specification cha | inge notice) | X | X | X | X |
| 4 | Code Protection and Security: Limitation of fla extra area and reset vector reprogramming (Schange notice) | ✓ | 1 | 1 | ✓ | |
| 5 | External Interrupt - Input Filter Timing Port F Shortened pulse rejection time | X | X | × | X | |
| 6 | External Interrupt - Input Filter Timing - All exterinput channels (except Port P07/ INTP6): Shor rejection time | | ✓ | / | ✓ | / |
| 7 | AFCAN: Sleep Mode Wakeup (Specification char | nge notice) | Х | X | X | X |
| 8 | On-Chip Debug Unit: Maximum input clock for Control Unit (DCU) is limited to 10 MHz | the Debug | X | Х | X | × |
| 9 | On-Chip Debug Unit: Execution of BROM-Startu be unintentionally interrupted at User S/W Addresses | | × | × | × | Х |
| 10 | CSIB: CSIB may stop operating | | Х | X | X | X |
| 11 | On-Chip Debug Unit: Reconfiguration of N-Wire N-Wire debug mode (µPD70F3427 only) | pins during | 1 | 1 | 1 | 1 |
| 12 | Flash Memory: Increased Power Consumption wooperating on the Sub Oscillator Clock (µPD70F34 | | √ | 1 | \ | ✓ |
| 13 | Flash Memory: Increased Power Consumption wooperating on the Sub Oscillator Clock (iRAM operations) | | X | X | X | X |
| 14 | Flash Memory: Flash Access Timing Violation a Clock Supply Switch | after System | X | X | X | X |
| 15 | CPU: HALT instruction (1) (μPD70F3426 only) (Schange notice) | Specification | 1 | 1 | 1 | ✓ |

| No. | Outline | µPD70F3421 | µPD70F3422 | µPD70F3423 | | |
|-----|--|----------------------|------------|------------|-----|-----|
| | | Rev. | 1.1 | 1.1 | 1.0 | 1.1 |
| | | Rank ^{Note} | I | Z | I | K |
| 16 | CPU: HALT instruction (2) (Specification change | 1 | 1 | 1 | ✓ | |
| 17 | DMA: MLE Bit Usage (Specification change notice | Х | X | Х | X | |
| 18 | Linear code execution across address 0x000 0x00008000 when the boot sectors are swapped use) | ✓ | 1 | / | 1 | |
| 19 | CPU: Speed limitation (1) (μPD70F3426 only) (Tetation) | echnical limi- | √ | 1 | / | 1 |
| 20 | CPU: Speed limitation (2) (μPD70F3427 only) (Tetation) | echnical limi- | √ | 1 | 1 | 1 |
| 21 | LCD-I/F: Reading the LBCTL0.TPF0 status flag may result in wrong information (Specification change notice) | | | Х | Х | Х |
| 22 | LCD-I/F: Writing to the LBDATA0W/ LBDATA0/register may lead to a corrupt data transfer (Tection) | | × | × | × | Х |

✓ : Not applicableX : Applicable

Note: The rank is indicated by the letter appearing at the 5th position from the left in the lot number, marked on each product.

(A.2) μPD70F3424, μPD70F3425

| No. | Outline | µPD70F3424 | | µPD70F3425 | | |
|-----|---|----------------------|----------|------------|-----|-----|
| | | Rev. | 1.0 | 1.1 | 1.1 | 2.0 |
| | | Rank ^{Note} | I | K | I | K |
| 1 | Flash Memory: Flash memory block operation Limitation) | s (Technical | X | X | × | ✓ |
| 2 | CPU: RAM mirror access (µPD70F3425 only) (schange notice) | Specification | √ | ✓ | × | X |
| 3 | Ports: PDSCn registers mirrors (Specification cha | ange notice) | X | X | X | X |
| 4 | Code Protection and Security: Limitation of fla extra area and reset vector reprogramming (schange notice) | 1 | 1 | × | ✓ | |
| 5 | External Interrupt - Input Filter Timing Port I Shortened pulse rejection time | P07/ INTP6: | X | X | × | ✓ |
| 6 | External Interrupt - Input Filter Timing - All exterinput channels (except Port P07/ INTP6): Short rejection time | | < | √ | × | X |
| 7 | AFCAN: Sleep Mode Wakeup (Specification char | nge notice) | X | Х | Х | X |
| 8 | On-Chip Debug Unit: Maximum input clock for Control Unit (DCU) is limited to 10 MHz | r the Debug | Х | Х | × | X |
| 9 | On-Chip Debug Unit: Execution of BROM-Startube unintentionally interrupted at User S/W Addresses | • | | × | × | X |
| 10 | CSIB: CSIB may stop operating | | Х | X | X | X |
| 11 | On-Chip Debug Unit: Reconfiguration of N-Wire N-Wire debug mode (µPD70F3427 only) | pins during | √ | √ | 1 | 1 |
| 12 | Flash Memory: Increased Power Consumption voperating on the Sub Oscillator Clock (µPD70F34 | | 1 | √ | 1 | 1 |
| 13 | Flash Memory: Increased Power Consumption voperating on the Sub Oscillator Clock (iRAM ope | | Х | Х | × | × |
| 14 | Flash Memory: Flash Access Timing Violation a Clock Supply Switch | after System | Х | X | × | × |
| 15 | CPU: HALT instruction (1) (µPD70F3426 only) (schange notice) | Specification | √ | √ | / | 1 |
| 16 | CPU: HALT instruction (2) (Specification change | notice) | X | X | X | X |

| No. | Outline | | µPD70F3424 | | µPD70F3425 | |
|-----|--|----------------------|------------|-----|------------|-----|
| | | Rev. | 1.0 | 1.1 | 1.1 | 2.0 |
| | | Rank ^{Note} | I | K | I | K |
| 17 | DMA: MLE Bit Usage (Specification change notice | Х | X | Х | Х | |
| 18 | Linear code execution across address 0x000 0x00008000 when the boot sectors are swapped use) | | × | × | × | |
| 19 | CPU: Speed limitation (1) (μPD70F3426 only) (Tetation) | echnical limi- | √ | / | 1 | / |
| 20 | CPU: Speed limitation (2) (μPD70F3427 only) (Tetation) | echnical limi- | 1 | / | 1 | 1 |
| 21 | LCD-I/F: Reading the LBCTL0.TPF0 status flag may result in wrong information (Specification change notice) | | | × | × | X |
| 22 | LCD-I/F: Writing to the LBDATA0W/ LBDATA0/register may lead to a corrupt data transfer (Tection) | | X | X | × | × |

✓ : Not applicableX : Applicable

Note: The rank is indicated by the letter appearing at the 5th position from the left in the lot number, marked on each product.

(A.3) μ PD70F3426, μ PD70F3426A, μ PD70F3427

| No. | Outline | | µPD70F3426 | рРD70F3426A | µPD70F3427 | |
|-----|--|----------------------|------------|-------------|------------|-----|
| | | Rev. | 1.0 | 1.0 | 1.0 | 1.1 |
| | | Rank ^{Note} | I | K | I | К |
| 1 | Flash Memory: Flash memory block operations Limitation) | s (Technical | 1 | 1 | 1 | 1 |
| 2 | CPU: RAM mirror access (µPD70F3425 only) (schange notice) | Specification | / | 1 | 1 | 1 |
| 3 | Ports: PDSCn registers mirrors (Specification cha | ange notice) | Х | Х | Х | X |
| 4 | Code Protection and Security: Limitation of flaextra area and reset vector reprogramming (schange notice) | ✓ | 1 | 1 | 1 | |
| 5 | External Interrupt - Input Filter Timing Port F Shortened pulse rejection time | P07/ INTP6: | √ | 1 | 1 | 1 |
| 6 | External Interrupt - Input Filter Timing - All exterinput channels (except Port P07/ INTP6): Shor rejection time | ✓ | 1 | 1 | 1 | |
| 7 | AFCAN: Sleep Mode Wakeup (Specification char | nge notice) | Х | Х | Х | X |
| 8 | On-Chip Debug Unit: Maximum input clock for Control Unit (DCU) is limited to 10 MHz | the Debug | Х | × | х | X |
| 9 | On-Chip Debug Unit: Execution of BROM-Startube unintentionally interrupted at User S/W Addresses | • | X | × | × | X |
| 10 | CSIB: CSIB may stop operating | | Х | Х | X | X |
| 11 | On-Chip Debug Unit: Reconfiguration of N-Wire N-Wire debug mode (µPD70F3427 only) | pins during | √ | 1 | × | × |
| 12 | Flash Memory: Increased Power Consumption voperating on the Sub Oscillator Clock (µPD70F34 | | Х | X | 1 | 1 |
| 13 | Flash Memory: Increased Power Consumption voperating on the Sub Oscillator Clock (iRAM operation) | | X | Х | × | Х |
| 14 | Flash Memory: Flash Access Timing Violation a Clock Supply Switch | after System | X | Х | х | X |
| 15 | CPU: HALT instruction (1) (μPD70F3426 only) (schange notice) | Specification | X | X | 1 | ✓ |
| 16 | CPU: HALT instruction (2) (Specification change | notice) | Х | Х | Х | X |

| No. | Outline | | µPD70F3426 | µРD70F3426A | µPD70F3427 | |
|-----|---|----------------------|------------|-------------|------------|----------|
| | | Rev. | 1.0 | 1.0 | 1.0 | 1.1 |
| | | Rank ^{Note} | I | K | I | K |
| 17 | DMA: MLE Bit Usage (Specification change notice | Х | Х | Х | Х | |
| 18 | Linear code execution across address 0x000 0x00008000 when the boot sectors are swapped (use) | | × | × | × | × |
| 19 | CPU: Speed limitation (1) (µPD70F3426 only) (Te tation) | chnical limi- | Х | 1 | 1 | 1 |
| 20 | CPU: Speed limitation (2) (µPD70F3427 only) (Te tation) | chnical limi- | √ | √ | × | √ |
| 21 | LCD-I/F: Reading the LBCTL0.TPF0 status flag may result in wrong information (Specification change notice) | | Х | Х | X | Х |
| 22 | LCD-I/F: Writing to the LBDATA0W/ LBDATA0/ LBDATA0I register may lead to a corrupt data transfer (Technical limitation) | | | × | × | × |

✓ : Not applicableX : Applicable

Note: The rank is indicated by the letter appearing at the 5th position from the left in the lot number, marked on each product.

(B) Description of Operating Precautions

No. 1 Flash Memory: Flash memory block operations (Technical Limitation)

Details

When calling flash operations based on blocks, e.g. "BlockErase", this operation is not executed correctly for certain combinations of start and end blocks.

This applies only to block combinations, where the area between the start block and the end block exceeds the 128 blocks boundary $7F_H$. For example, a block operation with start block = 10_H and end block = FF_H will fail, since the 128 blocks boundary $7F_H$ is exceeded.

Workaround

Flash block operations have to be separated into two steps such that the operations are separately executed within a single 128 blocks area 00_H to $7F_H$, respectively 80_H to FF_H .

Example by using functions of the self-programming library:

SelfLib_Erase(0x10,0xff) has to be separated into SelfLib_Erase(0x10,0x7f) and SelfLib_Erase(0x80, 0xff).

No. 2 CPU: RAM mirror access (µPD70F3425 only) (Specification change notice)

Details

The 32 KB RAM area $3FF 0000_H$ to $3FF 7FFF_H$ is mirrored to the 32 KB address range $3FF 8000_H$ to $3FF FFFF_H$. Since the upper 4 KB $3FF F000_H$ to $3FF FFFF_H$ overlaps the fixed peripheral I/O area, the RAM can not be accessed via this address range. Following an overview of the RAM areas and it's images:

| RAM | | RAM image | | |
|---|---------------|---|---------------|--|
| address range | accessibility | address range | accessibility | |
| 3FF 0000 _H - 3FF 3FFF _H | yes | 3FF 8000 _H - 3FF BFFF _H | yes | |
| 3FF 4000 _H - 3FF 6FFF _H | yes | 3FF C000 _H - 3FF EFFF _H | yes | |
| 3FF 7000 _H - 3FF 7FFF _H | yes | 3FF F000 _H - 3FF FFFF _H | no | |

Workaround

Do not access the RAM area 3FF $7000_{\rm H}$ – 3FF 7FFF $_{\rm H}$ via it's mirror area 3FF $6000_{\rm H}$ – 3FF FFFF $_{\rm H}$.

No. 3 Ports: PDSCn registers mirrors (Specification change notice)

Details

The port drive strength control registers PDSCn are mirrored to a second address. Following an overview of the PDSCEn register addresses:

| Register | Original address | Mirror address |
|----------|------------------------|------------------------|
| PDSC0 | FFFF F300 _H | FFFF F320 _H |
| PDSC1 | FFFF F3o2 _H | FFFF F322 _H |
| PDSC2 | FFFF F304 _H | FFFF F324 _H |
| PDSC3 | FFFF F306 _H | FFFF F326 _H |
| PDSC4 | FFFF F308 _H | FFFF F328 _H |
| PDSC5 | FFFF F3oA _H | FFFF F32A _H |
| PDSC6 | FFFF F3oC _H | FFFF F32C _H |
| PDSC7 | FFFF F3oE _H | FFFF F32E _H |
| PDSC8 | FFFF F310 _H | FFFF F330 _H |
| PDSC9 | FFFF F312 _H | FFFF F332 _H |
| PDSC10 | FFFF F314 _H | FFFF F334 _H |

Workaround

Though the PDSCn registers are accessible via both addresses, the original address is referenced as the only address. For instance, the mirror addresses will not be listed in the devices' header files.

No. 4 Code Protection and Security: Limitation of flash memory extra area and reset vector reprogramming (Specification change notice)

Details

The security function which limits reprogramming of the flash memory's extra area and reset vector via an external programmer is not provided with µPD70F3425 DS1.1.

Workaround

There is no workaround available.

No. 5 External Interrupt - Input Filter Timing Port P07/ INTP6: Shortened pulse rejection time

Details

The pulse rejection time for the analogue input filter of the port P07/ INTP6's external interrupt input filter is shorter than for the other external interrupt input channels. The effective pulse rejection time for the port P07/ INTP6 is:

min. 25,30 ns and max. 163,08 ns

Workaround

There is no workaround available.

No. 6 External Interrupt - Input Filter Timing - All external interrupt input channels (except Port P07/INTP6): Shortened pulse rejection time

Details

The pulse rejection time for the analogue input filter of the port P07/ INTP6's external interrupt input filter is shorter than for the other external interrupt input channels. The effective pulse rejection time for the port P07/ INTP6 is:

min. 25,30 ns and max. 163,08 ns

Workaround

There is no workaround available.

1. Description

When the AFCAN macro is set into SLEEP mode, it can be waken up by CAN bus activity. This waking up is asynchronous to the operation of the macro and the CPU. By configuration setting, a WAKEUP interrupt can be generated by the AFCAN macro on the wakeup event. While the interrupt is generated asynchronously, the AFCAN macro may need another dominant edge on the CAN bus, or software clearing of the SLEEP mode, in order to restart its synchronous operation.

During the time, after the interrupt already has been indicated, and before the CAN macro has restarted its synchronous operation, the registers of the AFCAN macro will not operate, because the AFCAN macro still remains in SLEEP mode. This time we will refer to as "wakeup dead time" in the following context.

To resolve from the wakeup dead time, software and/or hardware measures are required.

2. Exclusions

This Operating Precaution is **only applicable** to applications, which are fulfilling at least one of the following three conditions:

- SLEEP Mode of AFCAN is used **and** the possibility to wake up AFCAN by CAN-Bus events is given (see remark 1 below).
- During SLEEP mode of the AFCAN macro, a CAN-Bus wakeup condition occurs, while the AFCAN macro is supplied with clock (see remark 2 below) and
 - after waking up from SLEEP mode of the AFCAN macro, the application software does **not** wait until the SLEEP mode is released by polling the CnCTRL (PSMODE) register, before continuing operation with the AFCAN macro (see remark 3 below) **and**
 - the CPU can reach instructions, where AFCAN registers are accessed while the AFCAN macro is still in SLEEP mode, due to the missing waiting condition.
- During SLEEP mode of the AFCAN macro, a CAN-Bus wakeup condition occurs, while the AFCAN macro is supplied with clock (see remark 2 below) and
 - after waking up from SLEEP mode of the AFCAN macro, the CAN Bus Transceiver generates a **long-lasting or permanent dominant level** to the CRXD input of the AFCAN macro, instead of the propagated CAN-Bus level.

Remarks: 1. If the CAN-Bus Transceiver does not propagate the CAN-Bus signal, while the AFCAN macro is in SLEEP mode, and also does not forward a wakeup signal to CRXD, **this Operating Precaution is not applicable**.

- 2. The clock supply to the AFCAN macro can be stopped, depending on the features of the device, and the system design of the application. If the clock supply to the AFCAN macro is stopped, while a wakeup condition occurs, this Operating Precaution is not applicable.
- 3. The maximum waiting time for this loop can be up to 10 bits of the CAN-Bus Baudrate.

Waiting while retrying to clear CnINTS (Bit 5) can be used alternatively.

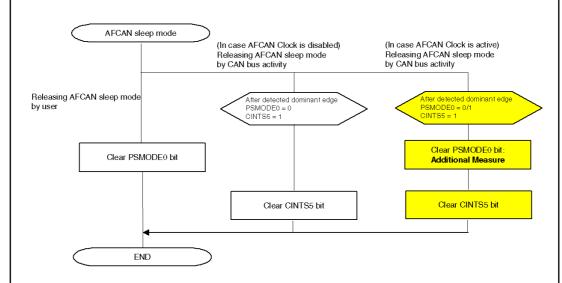
All other applications are not affected by this Operating Precaution.

3. Application Dependency

3.1 Overview

The following flowchart illustrates, how and whether additional measures have to be taken in software, to avoid the *wakeup dead time*.

Figure: Additional Measures in case AFCAN clock is active when waking up



3.2 Not affected Applications

3.2.1 Applications not using SLEEP mode

If SLEEP mode is not used, this Operating Precaution is not applicable.

3.2.2 Applications waking up from SLEEP mode by User Request only

If there is no condition, when SLEEP mode can be left by CAN-Bus activity, but only on User Request (by clearing the PSMODE flag by software), **this Operating Precaution is not applicable**.

3.2.3 Applications using a CPU Power Save Mode

If the clock to the AFCAN macro is disabled, while it is waken up from SLEEP mode, this operating precaution is not applicable.

This means, if the user selects a power save mode of the target device, which switches off the clock of the AFCAN macro, immediately after it had been set into SLEEP mode, like the CPU STOP mode, the precaution needs not to be considered.

This is associated with the software improvement hints below.

3.3 Affected Applications

3.3.1 Applications not waiting until SLEEP mode is left

If bus transceivers are used in conjunction with AFCAN, which will propagate the CAN bus signal to AFCAN permanently (not switched off or not in power saving modes), or, if bus transceivers are used in conjunction with AFCAN, which will propagate the unmodified CAN-Bus signal when waking up from a power save mode, the wakeup dead time lasts from the first recessive-to-dominant edge of the CAN-Bus signal, which generates the wake-up, until the next recessive-to-dominant edge of the CAN-Bus signal.

The worst case (maximum length) of the wakeup dead time, is given by the CAN bus speed and the rule of the CAN bus about the frequency of recessive-to-dominant edges. Given by the stuffing rule, at least every 10 bits, a recessive-to-dominant edge must occur.

If during the wakeup dead time, the CPU waits until the SLEEP mode is indicated to be cleared (either by polling the PSMODE flag, or by retrying to clear CnINTS[5]), **this operating precaution is not applicable**. In this case, the improvement hint according to 4.2.2 is followed implicitly.

If during the wakeup dead time, the CPU does not perform any access to the AFCAN macro in any case, **this operating precaution is not applicable**.

3.3.2 Applications using Bus Transceivers generating long-lasting dominant CAN-Bus Signals

If bus transceivers are used in conjunction with AFCAN, which *generate a permanent or long-lasting dominant level* when waking up from a power save mode, the operating precaution must be considered in any case.

In this case, the wakeup dead time lasts from the first recessive-to-dominant edge of the CAN bus signal, which generates the wake-up, until the next recessive-to-dominant edge of the CAN bus signal, depending on the behaviour of the CAN bus transceiver.

If no further dominant edge on the CAN bus occurs (in case of some CAN transceivers, which only provide one single edge on waking up), the time until SLEEP mode is left may become endless. Therefore, the waking up procedure of AFCAN regarding software, must be adjusted according to 4.1.1.

4. Software Improvement Hints

4.1 Recommended WAKEUP Handling by Software

4.1.1 Clearing the SLEEP Mode by Software

Within the WAKEUP interrupt routine, before accessing any other register or area of AFCAN, the SLEEP mode can be canceled by software, followed by a clearance of the WAKEUP interrupt flag.

Doing so, the AFCAN macro will start its synchronous operation right after these accesses. In the following C-code example, replace the objects in "<>" brackets by the hardware locations within your implementation. Use the appropriate access types, as described in the User's Manual.

```
WAKEUP INTERRUPT VECTOR -->
```

```
<CnCTRL_PSMODE> = 0;  /* Clear SLEEP Mode */
<CnINTS_CINTS5> = 1;  /* Clear INTS5 */
...
/* following other parts of interrupt routine */
```

•••

Remark: Clearing INTS5 is required to get another WAKEUP interrupt anyway, by specification.

4.2 Other WAKEUP Handling Hints

4.2.1 Switching off the Clock Supply to AFCAN, while in SLEEP Mode

If the clock supply to the AFCAN macro is stopped, while it is in SLEEP mode, the synchronisation of the WAKEUP works without any restriction. To achieve this, the documentation of clock controlling unit of the User's Manual must be consulted. Usually this is performed by setting the STOP/WATCH/Sub-WATCH mode of the CPU of the target device.

However, the user has to consider, that there must not be any WAKEUP condition (dominant level on the CAN-Bus), while the software is executing between setting SLEEP mode and stopping the AFCAN clock.

4.2.2 Using a Waiting Loop within the WAKEUP interrupt routine

Within the WAKEUP interrupt routine, create a waiting loop, which tests the capability of clearing the WAKEUP interrupt flag within AFCAN, by checking the actual power save mode. In the following C-code example, replace the objects in "<>" by the hardware locations within your implementation. Use the appropriate access types, as described in the User's Manual.

```
do
{
    AFCAN_SleepStatus = <CnCTRL_PSMODE>
    if( AFCAN_SleepStatus != 0 )
    {
        /* macro is still in SLEEP mode (waiting for latency time) */
        <CnINTS_CINTS5> = 1;    /* repeated trying to clear CINTS5 */
    }
} while( AFCAN_SleepStatus != 0 );
```

This improvement hint **cannot be applied**, if a CAN-Bus-Transceiver is attached to AFCAN, which generates a permanent or long-lasting dominant level to the FCRXDn receive input pin, if a wakeup condition occurs. Missing another dominant edge on the bus, the synchronisation will not happen, and the loop could run endlessly.

4.2.3 Using INIT Mode instead of SLEEP Mode

In this case, the waking up by CAN-bus activity must be performed via another free external interrupt. The CAN receive signal must be distributed on the FCRXDn pin, and to another external interrupt pin in parallel.

Using this external interrupt, the AFCAN macro can be restored into the previous operation mode.

This implementation will not use the SLEEP mode of AFCAN at all, and use the INIT mode instead.

No. 8 On-Chip Debug Unit: Maximum input clock for the Debug Control Unit (DCU) is limited to 10 MHz

Description

On-chip debugging is not possible in case that the input clock for the Debug Control Unit (DCU) has been set to 20 MHz.

Workaround

Implement all of the following measures:

- While debugging with any N-Wire based emulator, do limit the maximum input clock of the DCU to 10 MHz.
- Do consult the latest concerned tool documentation in order to verify that no OCD option is configured in your environment that will configure a higher DCU input clock than 10 MHz.
- Set the following environment variable: IE850 JCNDREGINIT=320

Special hint for 850Eserv:

Above mentioned configuration can be achieved by setting the On-Chip-Debugger (OCD) emulator option:"-2m".

Do never set the OCD emulator option:"-dck20", since it would configure a DCU input clock of 20 MHz.

No. 9 On-Chip Debug Unit: Execution of BROM-Startup Code can be unintentionally interrupted at User S/W Breakpoint Addresses

Description

If during program execution a Reset occurs (either from any internal Reset source or from the external Reset pin), the On-Chip Debugger may stop on the address of a previously configured S/W or H/W Breakpoint during the execution of the BROM's startup code. In this case, the following error message may be output:

"Couldn't read flash memory at xxx 0xc25 user system error (mask rom >area)"

Workaround

Implement all of the following measures when debugging with the On-Chip Debug Unit:

• Do not set any kind of breakpoint (either S/W or H/W Breakpoint) within any of the address ranges mentioned below:

0x0000 ... 0x000F 0x06D0 ... 0x2B23

Special hint.

Reserve the address range 0x06D0 ... 0x2B23 for constant data placement. In case using NEC's directive files, which are part of the Device-File package, this is the default assignment. If the program requires less constant data than that address space offers, modify the linker directive file in a way, that program code does not start before the address 0x2B24.

No. 10 CSIB: CSIB may stop operating

<u>Details - Master Mode Operation</u>

When any channel of CSIB is operated with a peripheral clock source different to the clock source of the CPU, the CSIB may stop operating. Depending on the CSIB operating configuration the CSIB behaves as described below.

- Transmit mode or transmit/receive mode:
 Any write to the related CBnTX0 register will no longer start a transmission sequence. Furthermore the related transmission interrupt request will not be generated.
- Receive mode:
 Any read from the related CBnRX0 register will no longer start a receive sequence.

 Furthermore the related receive interrupt request will not be generated.

The described CSIBn stuck condition can be escaped by initiating a system reset or by a sequential clear and set of the CBnCTL0.CBnPWR bit.

Details - Slave Mode Operation

When any channel of CSIB is operated in slave mode and an external clock signal is input via the SCKB*n* pin while no transmission or reception sequence is in progress the CSIB may stop operating. Depending on the CSIB operating configuration the CSIB behaves as described below.

- Transmit mode or transmit/receive mode
 Any further write to the CBnTX0 register followed by an external input clock signal input will no longer start a transmission sequence. Furthermore the related transmission interrupt request will not be generated.
- Receive mode:
 Any read from the related CBnRX0 register followed by an external input clock signal input will no longer start a receive sequence. Furthermore the related receive interrupt request will not be generated.

The described CSIBn stuck condition can be escaped by initiating a system reset or by a sequential clear and set of the CBnCTL0.CBnPWR bit.

Workaround - Master Mode Operation #1

In order to to avoid the CSIBn stuck condition in master mode use only the following CPU clock to CSIBn input clock combinations::

| CPU Clock Source | SPSEL0 | PERIC | SPCLK1 (BRGn) Source | CSIB Clock Input |
|------------------|--------|-------|----------------------|------------------|
| 4 MHz Main Osc | 0 | 0 | 4 MHz Main OSC | PCLK6 1, BRGn |
| | 0 | 1 | 4 MHz Main OSC | PCLK6 2, BRGn |
| | 1 | 0 | PLL | PCLK6 1 |
| | 1 | 1 | PLL | PCLK6 2 |
| PLL | 1 | 0 | PLL (Divided) | BRGn |
| | 1 | 1 | | PCLK1, BRGn |

No. 10 CSIB: CSIB may stop operating

(cont.)

Master Mode Operation - Workaround #2

In order to avoid the CSIBn stuck condition in master mode set bit CBnCTL1.CBnDAP to '1'. Pls. be aware of the changed CSIB communication type that comes along with this measure. By choosing workaround #2 no dedicated clock combination like in workaround #1 must be followed.

or

Master Mode Operation - Workaround #3 (Detection of CSIBn Stuck Condition)

In order to avoid a permanent stop of the CSIBn communication that may be caused by the CSIBn stuck limitation, the application software must be able to detect the CSIBn stuck condition by itself in order to restart the interrupted data transfer.

The stop of the CSIBn communication can be recognized by missing interrupt generation of INTCBnR or INTCBnT before the actual data transfer has finished (INTCBnR/ INTCBnT according to the selected operation mode Continuous/ Single transfer).

By choosing this method in order to detect a CSIBn stuck condition no dedicated clock combination as described in workaround #1 and no dedicated configuration of the bit CBnCTL1.CBnDAP as explained in workaround #2 must be followed.

Workaround - Slave Mode Operation

In order to avoid the CSIBn stuck condition in slave mode take the following precautions.

- Transmit mode or transmit/receive mode:
 Make sure the external CSIBn clock is not input in parallel when writing to the
 CBnTX0 register after a transmission sequence is finished.
- Receive mode:
 Make sure the external CSIBn clock is not input in parallel when reading from the
 CBnRX0 register after a reception sequence is finished.

No. 11 On-Chip Debug Unit: Reconfiguration of N-Wire pins during N-Wire debug mode (μPD70F3427 only)

Details

During N-Wire debug mode the configuration of the N-Wire interface pins can be changed by the concerned port mode register PM5. That register contents will not be ignored in case of an active N-Wire debug mode (OCDM0 = 1). As a consequence the application software is able to disable the N-Wire operation in case the concerned port mode register is enabling output configuration for the concerned N-Wire interface pins.

Workaround

The application software must check whether the device is operating in the N-wire debug mode or not. If it is operating in N-wire debug mode, any modifications to the related port mode register should be omitted. N-wire debug mode should be identified by checking whether the OCDM bit of the reset-controller is set and PPR05 is high.

No. 12 Flash Memory: Increased Power Consumption when CPU is operating on the Sub Oscillator Clock (μPD70F3426 only)

Details

The device μ PD70F3426 is equipped with two separate flash clusters. Each of those clusters does contain 1 MB of flash memory. One flash cluster is accessible via the VFB (V850 Fetch Bus, Address range: 0000 0000_H to 000F FFFF_H). The other flash cluster is accessible via the VSB (Address range: 0010 0000_H to 001F FFFF_H).

The Flash-cluster that is not accessed by CPU/DMAC when CPU starts operating on the Sub-Oscillator clock ($f_{CPU} = 32 \text{ kHz}$), will not enter the dedicated low-current operating mode that is normally active during that operating mode. As a consequence of that the device's current consumption will exceed the specified current consumption for this operating mode.

Workaround

In case the CPU need to operate using the Sub-Oscillator clock as CPU clock both of the device's flash clusters must be accessed at least once by CPU/DMAC in order to enable the concerned low-current operation mode for both flash clusters.

No. 13 Flash Memory: Increased Power Consumption when CPU is operating on the Sub Oscillator Clock (iRAM operation).

Details

In case the CPU starts operation using the Sub-Oscillator clock as CPU clock ($f_{CPU}=32$ kHz) and is only executing instructions and reading data that are not located within the internal flash-memory (so the accessed instructions/ data are located within the internal RAM or external memory in case of μ PD70F3427) the internal flash-memory will not enter the dedicated low-current operating mode. As a consequence of that the device's current consumption will exceed the specified current consumption in this operating mode.

Workaround

In case the CPU need to start operation using the Sub-Oscillator clock as CPU clock supply ($f_{CPU} = 32 \text{ kHz}$) and furthermore the CPU does not apply any access to its internal flash-memory, the flash memory must be accessed at least once by CPU/DMAC in order to enable the low-current operating mode of the flash memory.

For the derivative μ PD70F3426 both of the device's flash clusters must be accessed at least once by CPU/DMAC in order to enable the concerned low-current operation mode for both flash clusters.

No. 14 Flash Memory: Flash Access Timing Violation after System Clock Supply Switch

Details

In case the system clock (f_{CPU}) is changed from the clock of the Sub-Oscillator ($f_{CPU} = 32$ kHz) to the clock of the Main-Oscillator or to any clock of SSCG or PLL, a CPU access to the internal flash-memory may violate the flash-timing. As a consequence to that the concerned read data respectively the read instruction code may be corrupted.

This behaviour will appear in case one of the below described sequences is applied from the application:

Precondition

The CPU operates using the sub-oscillator as the clock source ($f_{CPU} = 32$ kHz). As a consequence of that the flash memory is operating in the low-current operating mode (In case of μ PD70F3426 it's assumed that both flash-clusters are operating in the low-current operating mode).

- (1A) The system clock (f_{CPU}) that is the clock of the Sub-Oscillator (f_{CPU} = 32 kHz) is switched to another clock-source (Main-Oscillator or any clock of the SSCG or PLL) by immediate configuration of PCC.
- (2A) A read/fetched data/instruction is corrupted due to a timing violation on the flash-cluster.

or

- (1B) The CPU enters the STOP standby mode.
- (2B) After the standby mode has been released by an appropriate wake-up event, the CPU starts operation using the Main-Oscillator clock as the system clock (f_{CPU} = 4 MHz) after having passed the oscillator stabilization time. This is the default configuration after having been released from the STOP mode.
- (3B) A read/fetched data/instruction is corrupted due to a timing violation on the flash-cluster.

or

(1C) The CPU enables and enters the Sub-WATCH standby mode with the following configuration of the Power save mode register PSM:

PSM.OSCDIS = 0 // Main Oscillator enabled

- (2C) After the standby mode has been released by an appropriate wake-up event, the CPU starts operation using the Main-Oscillator clock as the system clock (f_{CPU} = 4 MHz) after having passed the oscillator stabilization time. This is according to the configuration of the Power save mode register PSM: PSM.OSCDIS = 0.
- (3C) The read/fetched data/instruction is corrupted due to a timing violation on the flashcluster.

No. 14 Flash Memory: Flash Access Timing Violation after System Clock Supply Switch

(cont.)

Workaround

For any application requesting a switch of the system clock from Sub-Oscillator clock (f_{CPU} = 32 kHz) to any other clock (e.g. Main-Oscillator or SSCG or PLL) do implement all of the following measures:

- Do never directly change the system clock from Sub-Oscillator clock (f_{CPU} = 32 kHz) to the clock of the Main-Oscillator or any clock of the SSCG or PLL by executing the concerned instructions from the internal flash-memory.
- In case the application requests to enter the Sub-Watch-Mode:
- Do never enter the Sub-WATCH mode with the following configuration of the Power save mode register PSM:

```
PSM.OSCDIS = 0 // Main Oscillator enabled
```

 Do always enter the Sub-WATCH mode with the following configuration of the Power save mode register PSM:

```
PSM.OSCDIS = 1 // Main Oscillator disabled
```

 After the Sub-WATCH standby-mode has been released by an appropriate wake-up event and furthermore the application requires another system clock (f_{CPU}) than the clock of the Sub-Oscillator, first do enable the Main-Oscillator by clearing the bit OSCDIS of the Power Save Mode register PSM:

```
PSM.OSCDIS = 0 // Main Oscillator enabled
```

- Do wait until the Main-Oscillator has stabilized by verifying the status flag OSCSTAT in the Clock Generator Status register CGSTAT.
- In case any clock of the SSCG or PLL should be used as the system clock (f_{CPU}), do
 enable the desired clock source and wait for the specified stabilization time.
- The instructions that will configure the system clock (f_{CPU}) must be executed from the internal RAM. Do not execute any of those configurations by executing the concerned instructions from the internal flash memory:
 - As the consequence, do jump to the internal RAM memory and execute the concerned instructions from this memory location.
 - Disable the generation of any maskable or non-maskable interrupt. This is achieved with the following configuration of the processor's Program Status Word PSW:

```
PSW.ID = 1 // Disables maskable interrupts
PSW.NP = 1 // Disables non-maskable interrupts
```

 Do configure the desired clock source as system clock (f_{CPU}) by configuring the Processor clock control register PCC (PCC.CKS0 and PCC.CKS1).

No. 14 Flash Memory: Flash Access Timing Violation after System Clock Supply Switch

(cont.)

- Do apply a dummy read access to the flash-cluster (For the derivative μPD70F3426, do apply a dummy read access to each of both flash-clusters). As a consequence to that the flash-cluster is enabling high-speed operation mode.
- Do wait for at least 28.5 µs before applying any new access to the internal flash memory. This wait time can be realized by execution of an appropriate software loop.
- With regards to the application needs, do enable the generation of maskable and non-maskable interrupts. This can be achieved with the following configuration of the processor's Program Status Word PSW:

```
PSW.ID = 0 // Enables maskable interrupts
PSW.NP = 0 // Enables non-maskable interrupts
```

 Do jump back to the internal flash-memory and continue execution of the application software.

Workaround

In case the application requires that the CPU enters the STOP standby mode, do never enter the STOP standby mode in case the clock of the Sub-oscillator is used as the system clock ($f_{CPU} = 32 \text{ kHz}$).

No. 15 CPU: HALT instruction (1) (µPD70F3426 only) (Specification change notice)

Details

Data read from upper 1 MB code flash (internal VSB flash memory located from 00100000H to 001FFFFFH) becomes invalid when a HALT instruction is executed immediately afterwards.

Workaround

To avoid the critical situation both following conditions must be applied:

- The last 4 instructions before the halt execution may not perform a data access to the upper 1 MB code flash (internal VSB flash memory located from 00100000H to 001FFFFFH).
- HALT instructions may be only executed from lower 1 MB code flash (internal VFB flash memory located from 00000000H to 000FFFFFH).

No. 16 CPU: HALT instruction (2) (Specification change notice)

Details

When the cpu executes a HALT instruction the successive output of the flash memory may become undefined. The undefined data passes through the data latches, that are transparent and finally will be fetched by the cpu as sequential instruction code.

Workaround

Avoid sequential code execution after HALT instruction. This can be achieved by one of the following measures:

(1) Non-sequential code execution due to ISR execution after HALT instruction

The HALT mode may be only released with enabled interrupts (EI) by a non masked maskable interrupt request or a non-maskable interrupt request (NMI). In this case the cpu does not execute the instructions located after the HALT instruction, but does an initial access to the interrupt handler address.

Note: Take care that no interrupt request is generated that is masked (when the corresponding xxMK bit is set to 1 in the xxIC registers), since the cpu would not branch to the interrupt handler address in that case.

(2) Branch after HALT instruction

Place a branch immediately after the HALT instruction, that is fetched before the cluster is going to standby and executed when standby is released.

Example:

```
br EvenAligned
nop
nop

.align 8
EvenAligned:

   halt
   br UserCode
   nop
   nop
   nop
.align 8
UserCode:
```

Note: "align 8" means placing the code to the even flash memory addresses (0x...0 and 0x...8).

No. 17 DMA: MLE Bit Usage (Specification change notice)

Details

Do not modify the setting of the MLEn bit in the DCHCn register while the DMA channel n is activated and DMA transfers of channel n are executed in the background.

Modify the MLE*n* bit when the corresponding channel n is one of the following periods. (The operation is not guaranteed if modified at another timing).

- Time from system reset to the generation of the first DMA transfer request of channel n
- Time from DMA transfer end while MLEn=0 (after terminal count) to the generation of the next DMA transfer request
- Time from the forcible termination (after the INITn bit has been set to 1) to the generation
 of the next DMA transfer request

The DMA channel transfer may stuck if the MLEn bit is modified. It may continue at the next DMA transfer request when (dummy) read accesses of the DCHCn register are performed to clear the TCn bit. In this case, DMA transfers may be lost if (dummy) read accesses are performed too late.

Workaround

If the application require to clear the MLE*n* bit while DMA transfers are executed in the background, following workaround must be applied:

The workaround is independent of VSWC register setting, NPB access retries, code location or the use of other DMA channels.

Four successive dummy read accesses to the DCHCn register should be performed directly before the MLEn bit is cleared. This sequence must not be disrupted or interrupted. To avoid a possible interrupt, the interrupt disable state should be entered by the 'di' instruction. After clearing the MLEn bit, the previous interrupt enable state may be restored.

The workaround should be written in (inline-) assembler to avoid disturbing the register access sequence.

```
di  // Avoid interruption of following sequence

Id.b DCHCn_ADR[r0], r0
Id.b DCHCn_ADR[r0], r0
Id.b DCHCn_ADR[r0], r0
Id.b DCHCn_ADR[r0], r0
Id.b DCHCn_ADR[r0]  // Clear MLEn Bit

ei  // Restore previous interrupt enable state
```

No. 18 Linear code execution across address 0x00010000 and 0x00008000 when the boot sectors are swapped (Direction of use)

Details

When the boot sectors of the flash memory are swapped, make sure that no linear code execution across the addresses 0x00010000 or 0x00008000 is done, because this can lead to unexpected behaviour.

Workaround

None.

No. 19 CPU: Speed limitation (1) (μPD70F3426 only) (Technical limitation)

Details

The CPU clock VBCLK of μ PD70F3426 must not be set above 48 MHz +/- 5% typical frequency modulation range of the SSCG dithering function, if a peripheral clock base (SPCLK0) of 16 MHz must be maintained.

If necessary, the given limit can be extended to 56 MHz +/- 5% typical frequency modulation range of the SSCG dithering function.

Workaround

There is no workaround available.

No. 20 CPU: Speed limitation (2) (µPD70F3427 only) (Technical limitation)

Details

The CPU clock VBCLK of μ PD70F3427 must not be set above 48 MHz +/- 5% typical frequency modulation range of the SSCG dithering function.

Workaround

There is no workaround available.

No. 21 LCD-I/F: Reading the LBCTL0.TPF0 status flag may result in wrong information (Specification change notice)

<u>Details</u>

The LBCTL0 register of the LCD bus interface (LCD-I/F) provides the status bit TPF0 that indicates whether

"0: The (external) LCD bus interface is idle" or

"1: Data is transferred on the (external) LCD bus interface".

Reading the TPF0 flag may indicate by accident a wrong status.

Workaround

None.

Instead of polling the TPF0 flag, use an interrupt procedure (INTLCD) or a DMA transfer. Polling of the IF flag of the corresponding interrupt control register is not affected and can be applied as well.

No. 22 LCD-I/F: Writing to the LBDATA0W/ LBDATA0/ LBDATA0L register may lead to a corrupt data transfer

(Technical limitation)

Details

When writing to the LBDATA0x register while a transfer on the LCD data bus is ongoing a corrupt data transfer may be the result. The critical situation can occur under certain clock constellations.

Workaround

The application software may choose one of the following workarounds:

- a) Do not write to the LBDATA0x register while a transfer is ongoing. To ensure this operate the LBDATA0x register upon the occurrence of the LCD Bus Interface interrupt INTLCD with LBCTL0.TCIS0 set to 1.
- b) Use one of the following clock setups in order to avoid the failure mode. The following markers are used.
- ✓ : valid clock combination
- x: critical clock combination, where the LBDATA0x register may be captured incorrectly
- : combination, which is invalid since it injures the specification (LCD bus clock must be less or equal 16 MHz) is marked by orange filling.
- combination, which is invalid only above 48 MHz is marked by yellow filling.
- When CPU system clock (VBCLK) and LCD bus clock are supplied by PLL.

μPD70F3421/22/23/24

| PLLCLK = 32 MHz | D | CPU System Clock (VBCLK) | | |
|--|----------------------|--------------------------|------------------|--|
| SCC.SPSEL[1:0] = 01 SPCLK0 = PLLCLK/2 | SPCLK1 = PLLCLK/4 | 32 MHz | 16 MHz | |
| LCD Bus Clock | Configuration | PCC.CKS[1:0]=11B | PCC.CKS[1:0]=10B | |
| 16 MHz | LBCTL0.LBC0[1:0]=00B | ✓ | Х | |
| 8 MHz | LBCTL0.LBC0[1:0]=01B | Х | Х | |

μPD70F3425/26/26A

| PLLCLK = 32 MHz | D | CPU System Clock (VBCLK) | | |
|---|----------------------|--------------------------|------------------|--|
| SCC.SPSEL[1:0] = 01 SPCLK0 = PLLCLK/2, | SPCLK1 = PLLCLK/4 | 32 MHz | 16 MHz | |
| LCD Bus Clock | Configuration | PCC.CKS[1:0]=11B | PCC.CKS[1:0]=10B | |
| 16 MHz | LBCTL0.LBC0[1:0]=00B | X | ✓ | |
| 8 MHz | LBCTL0.LBC0[1:0]=01B | Х | ✓ | |

μPD70F3427

| PLLCLK = 32 MHz | _ | CPU System Clock (VBCLK) | | | | | | |
|--|--------------------------|--------------------------|------------------|--|--|--|--|--|
| SCC.SPSEL[1:0] = 01 SPCLK0 = PLLCLK/2 | В , SPCLK1 = PLLCLK/4 | 32 MHz | 16 MHz | | | | | |
| LCD Bus Clock | Configuration | PCC.CKS[1:0]=11B | PCC.CKS[1:0]=10B | | | | | |
| 16 MHz | LBCTL0.LBC0[1:0]=00B | X | Х | | | | | |
| 8 MHz | LBCTL0.LBC0[1:0]=01B | Х | X | | | | | |

No. 22 LCD-I/F: Writing to the LBDATA0W/ LBDATA0/ LBDATA0L register may lead to a corrupt data transfer (Technical limitation)

When CPU system clock (VBCLK) and LCD bus clock are supplied by SSCG.

μPD70F3421/22/23/24

| × | | | ٥ | CPU System Clock (VBCLK) | (CLR) |
|--|------------------|--------------------|------------------------|--------------------------|--|
| × | Configuration | ıration | SSCLK | SSCLK/3 | SSCLK /2, /4/, /6, /8 |
| 2.0000 | LBCTL0.LBC0[1:0] | SCPS.SPSPS[2:0] | SCPS.VBSPS[2:0] = 000B | SCPS.VBSPS[2:0] = 010B | SCPS.VBSPS[2:0] = {001B, 011B, 101B, 111B} |
| SPCLK0 = SSCCLK | | 000B | 1 | / | × |
| SPCLK0 /2 = SSCCLK /2 | | 001B | / | / | × |
| SPCLKO /3 = SSCCLK /3 00B | | 010B | / | / | × |
| SPCLK0 /4, /6, /8 = SSCCLK /4, /6, /8 | | {011B, 101B, 111B} | × | × | > |
| SPCLK1 = SSCCLK /2 | | 000B | / | / | × |
| SPCLK1 /2 = SSCCLK /4 | | 001B | × | × | > |
| SPCLK1 /3 = SSCCLK /6 01B | | 010B | × | × | > |
| SPCLK1 /4, /6, /8 = SSCCLK /8, /12, /16 | | {011B, 101B, 111B} | × | × | ` |
| SPCLK2 = SSCCLK /4 | | 000B | × | × | > |
| SPCLK2 /2 = SSCCLK /8 | | 001B | × | × | > |
| SPCLK2 /3 = SSCCLK /12 10B | | 010B | × | × | > |
| SPCLK2 /4, /6, /8 = SSCCLK /16, /24, /32 | | {011B, 101B, 111B} | × | × | > |
| SPCLK5 = SSCCLK /32 | | 8000 | × | × | <i>></i> |
| SPCLK5 /2 = SSCCLK /46 | | 001B | × | × | > |
| SPCLK5 /3 = SSCCLK /96 11B | | 010B | × | × | > |
| SPCLK5 /4, /6, /8 = SSCCLK /128, /192, /256 | | {011B, 101B, 111B} | X | × | , |

No. 22 LCD-I/F: Writing to the LBDATA0W/ LBDATA0/ LBDATA0L register may lead to a corrupt data transfer (Technical limitation)

| (T | echni | hnical limitation) | | | | | | | | | | | | | | | | | | | |
|----|---|-----------------------|--|-----------------|-----------------------|--------|--|--------------------|-----------------------|-----------------------|--|--------------------|-----------------------|----------------------|---|---------------------|------------------------|-------------|--|--|--|
| μF | PD70 | F34 | 125 | | | | | | | | | | | | | | | | | | |
| | CLK) | SSCLK /2, /4/, /6, /8 | SCPS.VBSPS[2:0] = {001B, 011B, 101B, 111B} | <i>/</i> | / | / | > | <i>A</i> | > | > | ` | × | × | × | × | × | × | × | × | | |
| | CPU System Clock (VBCLK) | SSCLK /3 | SCPS.VBSPS[2:0] = 010B | × | × | × | × | × | × | × | × | X | / | > | > | × | > | \ <u></u> | / | | |
| | | SSCLK | SCPS.VBSPS[2:0] = 000B | × | × | × | × | × | × | × | × | X | / | ^ | > | × | ^ | <i>></i> | / | | |
| | | Configuration | SCPS.SPSPS[2:0] | 000B | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | | |
| | C.CKS[1:0] = 01B | Config | LBCTL0.LBC0[1:0] | | | 00B | | | | 01B | | | | 10B | | | | 118 | | | |
| | SSCCLK = 48 MHz, 64 MHz SCC.SPSEL[1:0] = 11B; PCC.CKS[1:0] = 01B | | LCD Bus Clock | SPCLK0 = SSCCLK | SPCLK0 /2 = SSCCLK /2 | CLK /3 | SPCLK0 /4, /6, /8 = SSCCLK /4, /6, /8 | SPCLK1 = SSCCLK /2 | SPCLK1 /2 = SSCCLK /4 | SPCLK1 /3 = SSCCLK /6 | SPCLK1 /4, /6, /8 = SSCCLK /8, /12, /16 | SPCLK2 = SSCCLK /4 | SPCLK2 /2 = SSCCLK /8 | SPCLK2/3 = SSCCLK/12 | SPCLK2 /4, /6, /8 = SSCCLK /16, /24, /32 | SPCLK5 = SSCCLK /32 | SPCLK5 /2 = SSCCLK /46 | (0 | SPCLK5 /4, /6, /8 = SSCCLK /128, /192, /256 | | |

No. 22 LCD-I/F: Writing to the LBDATA0W/ LBDATA0/ LBDATA0L register may lead to a corrupt data transfer (Technical limitation)

| | PD70F3426/26A | | | | | | | | | | | | | | | | | | | |
|--------------------------|--|-----------------------|--|-----------------|-----------------------|-----------------------|--|--------------------|-----------------------|-----------------------|--|--------------------|-----------------------|------------------------|---|---------------------|------------------------|-------------|--|--|
| | _ | | | | | | | | | | | | | | | | | | | |
| ICLK) | | SSCLK /2, /4/, /6, /8 | SCPS.VBSPS[2:0] = {001B, 011B, 101B, 111B} | <i>/</i> | <i>/</i> | / | > | 1 | > | > | > | × | × | × | × | × | × | × | × | |
| CPU System Clock (VBCLK) | • | SSCLK/3 | SCPS.VBSPS[2:0] = 010B | × | × | × | × | × | × | × | × | / | > | ^ | > | <i>></i> | > | > | > | |
| S | | SSCLK | SCPS.VBSPS[2:0] = 000B | × | × | × | × | × | × | × | × | / | <i>></i> | <i>></i> | > | <i>></i> | <i>></i> | <i>></i> | > | |
| | | ıration | SCPS.SPSPS[2:0] | 8000 | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | |
| | C.CKS[1:0] = 01B | Configuration | LBCTL0.LBC0[1:0] | | | 00B | | | | 01B | | | | 10B | | | | 118 | | |
| SSCCLK = 48 MHz, 64 MHz | SCC.SPSEL[1:0] = 11B; PCC.CKS[1:0] = 01B | | LCD Bus Clock | SPCLK0 = SSCCLK | SPCLK0 /2 = SSCCLK /2 | SPCLK0 /3 = SSCCLK /3 | SPCLK0 /4, /6, /8 = SSCCLK /4, /6, /8 | SPCLK1 = SSCCLK /2 | SPCLK1 /2 = SSCCLK /4 | SPCLK1 /3 = SSCCLK /6 | SPCLK1 /4, /6, /8 = SSCCLK /8, /12, /16 | SPCLK2 = SSCCLK /4 | SPCLK2 /2 = SSCCLK /8 | SPCLK2 /3 = SSCCLK /12 | SPCLK2 /4, /6, /8 = SSCCLK /16, /24, /32 | SPCLK5 = SSCCLK /32 | SPCLK5 /2 = SSCCLK /46 | 36/ XT | SPCLK5 /4, /6, /8 = SSCCLK /128, /192, /256 | |

No. 22 LCD-I/F: Writing to the LBDATA0W/ LBDATA0/ LBDATA0L register may lead to a corrupt data transfer (Technical limitation)

| (T | echni | hnical limitation) | | | | | | | | | | | | | | | | | | | |
|----|---|-----------------------|--|-----------------|-----------------------|--------|--|--------------------|-----------------------|-----------------------|--|--------------------|-----------------------|-----------------------|---|---------------------|------------------------|------|--|--|--|
| μF | PD70 | F34 | 127 | | | | | | | | | | | | | | | | | | |
| | CLK) | SSCLK /2, /4/, /6, /8 | SCPS.VBSPS[2:0] = {001B, 011B, 101B, 111B} | / | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | | |
| | CPU System Clock (VBCLK) | SSCLK /3 | SCPS.VBSPS[2:0] = 010B | × | × | × | X | × | × | × | × | X | × | × | × | × | × | × | × | | |
| | _ | SSCLK | SCPS.VBSPS[2:0] = 000B | × | × | × | × | × | × | × | × | X | × | × | × | × | × | × | × | | |
| | | ıration | SCPS.SPSPS[2:0] | 000B | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | 000B | 001B | 010B | {011B, 101B, 111B} | | |
| | C.CKS[1:0] = 01B | Configuration | LBCTL0.LBC0[1:0] | | | 00B | | | | 01B | | | | 10B | | | • | 118 | | | |
| | SSCCLK = 48 MHz, 64 MHz SCC.SPSEL[1:0] = 11B; PCC.CKS[1:0] = 01B | | LCD Bus Clock | SPCLK0 = SSCCLK | SPCLK0 /2 = SSCCLK /2 | CLK /3 | SPCLK0 /4, /6, /8 = SSCCLK /4, /6, /8 | SPCLK1 = SSCCLK /2 | SPCLK1 /2 = SSCCLK /4 | SPCLK1 /3 = SSCCLK /6 | SPCLK1 /4, /6, /8 = SSCCLK /8, /12, /16 | SPCLK2 = SSCCLK /4 | SPCLK2 /2 = SSCCLK /8 | SPCLK2/3 = SSCCLK /12 | SPCLK2 /4, /6, /8 = SSCCLK /16, /24, /32 | SPCLK5 = SSCCLK /32 | SPCLK5 /2 = SSCCLK /46 | (0 | SPCLK5 /4, /6, /8 = SSCCLK /128, /192, /256 | | |

(C) Valid Specification

| Item | Date published | Document No. | Document Title |
|------|----------------|-----------------|--|
| 1 | February 2004 | U14559EJ3V1UM00 | V850E1 32-Bit Microprocessor Core Architecture (User's Manual) |
| 2 | Oct 21, 2010 | R01UH0129ED0601 | V850E/Dx3 - DJ3/DL3 Hardware (User's Manual) |
| 3 | Jun 1, 2011 | R01DS0049ED0210 | V850E/Dx3 - DJ3/DL3 (Data Sheet) |

(D) Revision History

| Item | Date published | Document No. | Comment |
|------|----------------|------------------|--|
| 1 | October 2006 | EASE-LL-0003-0.1 | First release |
| 2 | February 2007 | EASE-LL-0003-0.2 | Operating precautions no. 11 to 16 were added. |
| 3 | May 2007 | EASE-LL-0003-0.3 | Compiler information on operating precautions no. 2 and 3 was updated. Operating precaution no. 17 was added. |
| 4 | November 2007 | U19052EE1V0IF00 | Operating precautions no. 1 to 3 of former document (EASE-LL-0003-0.3) were removed since these are already described in the valid specification. The remaining operating precautions were renumbered from 1 to 14. Operating precautions no. 15 and 16 were added. |
| 5 | May 2008 | U19052EE1V1IF00 | Operating precaution no. 17 was added. |
| 6 | December 2008 | U19052EE2V0IF00 | Operating precaution no. 18 was added |
| 7 | Jan 19, 2011 | R01TU0012ED0300 | Operating precautions no. 19 and 20 were added |
| 8 | Feb 27, 2012 | R01TU0012ED0400 | Markers of operating precaution no. 4 in tables of operating precautions (A.1) and (A.2) were added (only μPD70F3425, DS1.1 was affected). Operating precautions no. 21 and 22 were added. |

